



# Obsah

Úvod.....	3
Co je to regulární výraz?.....	3
K čemu jsou regulární výrazy v Mergadu užitečné?.....	3
Nejčastěji používané značky a jejich využití.....	5
Tečka .....	5
Hvězdička *.....	5
Otazník ?.....	5
Plus +.....	5
Hranaté závorky [].....	6
Kulaté závorky ().....	6
Složené závorky {}.....	7
Stříška ^.....	7
Dolar \$.....	8
Svislítko  .....	8
Znak s se zpětným lomítkem \s.....	9
Negace.....	9
\D a \S.....	9
Přehled značek.....	10
Příklady využití regulárních výrazů v Mergadu.....	11
EAN.....	11
Hromadné odstranění EAN kódů z popisku produktů.....	11
Kontrola EANu.....	12
Odebrání velikosti z PRODUCTNAME.....	13
Složitější příklady využití regulárních výrazů.....	14
Přepsání informace z jednoho elementu do druhého.....	14
Duplicitní slova v elementu.....	15
Závěrem.....	15
Doporučené zdroje.....	15
Test na závěr.....	16
Výsledky testu.....	17

# Úvod

Jistě jste se už na Mergadově blogu setkali s magickým odkazem na regulární výrazy. Řetězce znaků s podivnými značkami sice nepatří do základních znalostí, ale sami uvidíte, jak moc se vám budou hodit. Proniknout do jejich tajemství se vám bude snažit usnadnit tento dokument.

Na začátek se budeme krátce věnovat teorii, dále si představíme jednotlivé značky, ze kterých jsou regulární výrazy složeny, a ve druhé polovině představíme konkrétní příklady použití v Mergadu. Úplný závěr potom patří krátkému testu, na kterém si můžete svoje nově nabyté znalosti otestovat.

## Co je to regulární výraz?

Regulární výraz si můžete představit jako šikovného vyhledávacího pomocníka, který vám přijde na pomoc, když **nevíte, co přesně v textu nebo kódu hledáte**, ale **víte, jak zhruba to má vypadat**.



Tedy pokud například neznáte konkrétní číselnou kombinaci, kterou výraz obsahuje, ale víte, že má tato kombinace právě 8 číslic, můžete použít regulární výraz. Výsledkem vyhledávání pak budou všechny části textu, které odpovídají zadaným podmínkám. **Regulární výraz je tak textový řetězec, který pasuje na množinu řetězců na základě určitých syntaktických pravidel.**

## K čemu jsou regulární výrazy v Mergadu užitečné?

Bez používání regulárních výrazů se v Mergadu nejspíš nějakou chvíli obejdete, jak jim ale přijdete na kloub, už je nebudete chtít opustit – výrazně totiž usnadňují práci a šetří čas při **hromadných úpravách dat**. Problém, který byste museli řešit mnoha výběry, zvládnete s jedním regulárním výrazem.

Regulární výrazy využijete v Mergadu například:

- při **vytváření výběrů a vyhledávání** (na kartě Produkty -> Jednoduché rozhraní vyhledávání -> obsahuje/neobsahuje regulární výraz, obdobně lze regulární výrazy používat v Rozšířeném rozhraní vyhledávání);

- při vytváření proměnných;
- u **pravidel typu najít a nahradit** (třeba odstranit velikost výrobku z PRODUCTNAME);
- **kontroly výrazů** – zda má EAN počet znaků, který má mít.

Než se ale vhrneme na vyhledávání a upravování, považujeme za důležité vysvětlit vám, jakým způsobem se regulární výrazy tvoří a co který použitý symbol znamená.

# Nejčastěji používané značky a jejich využití

Regulární výrazy vyhledávají text pomocí nejrůznějších značek, přičemž základní z nich nyní představíme.

## Tečka .

Skrze tečku hledáme takové řetězce, u nichž **právě jeden znak** neznáme. Použijeme v případě, kdy:

- chceme najít všechny produkty, jejichž **popis obsahuje slovo "diskuze"** a nejsme si jistí, zda někde nebude "diskuse", použijeme **regulární výraz disku.e** (tomu vyhoví: diskuze i diskuse)
- chceme najít všechny produkty, jejichž **název obsahuje tři čísla**: 1 na začátku, 3 na konci a s tím, že prostřední číslo neznáme (může být libovolné nebo může jít o jakýkoliv znak), použijeme **regulární výraz 1.3** (tomu vyhoví: 103, 113, 123, 133, atd.)

## Hvězdička \*

Hvězdičkou vyhledáme výrazy, které obsahují **konkrétní znak 0 až nekonečně krát**. Použijeme ji v případě, kdy:

- chceme najít všechny produkty, jejichž název obsahuje **Baterie a volitelně také označení velikosti baterií** (A, AA nebo AAA), použijeme regulární výraz: **Baterie A\*** (tomuto vyhoví všechny následující tvary: "Baterie ", "Baterie A", "Baterie AA", "Baterie AAA")

## Otazník ?

Pomocí otazníku najdeme výraz, ve kterém se náš znak nevyskytuje nebo se vyskytuje jen jednou. Použijeme ho tedy v případě, kdy:

- chceme najít všechny produkty, které mají **v názvu velikost L nebo velikost XL** a přitom už nechceme najít XXL ani XXXL, použijeme regulární výraz: **Velikost X?L** (tomuto vyhoví všechny následující tvary: Velikost L, Velikost XL, tedy "X" tam nemusí být přítomno vůbec nebo maximálně jedenkrát)

## Plus +

Plus funguje podobně jako hvězdička, vyhledává ale takové řetězce, ve kterých je **znak alespoň jednou zastoupen**. Používáme ho v případě, kdy:



- chceme najít **všechny produkty ve velikosti XL, XXL a XXXL**, použijeme regulární výraz **X+L** (tedy "X" tam musí být přítomno minimálně jedenkrát, maximálně to není neomezeno)

Tip: v Mergadu u regulárních výrazů **nezáleží na tom**, zda hledáte text zadaný malými nebo velkými písmeny. Když budete hledat slovo "Velikost", najde se: "Velikost", "velikost" nebo třeba i "VeLiKoSt".

## Hranaté závorky []

Hranaté závorky pomohou s výběrem takového textu, ve kterém se vyskytuje **jeden ze skupiny znaků**, kterou zadáme, tedy například:

- Chceme najít všechny produkty, které mají v názvu **nějaké číslo**, použijeme regulární výraz **[0-9]** (alternativou je potom použití výrazu `\d`, který v regulárních výrazech představuje jakoukoli číslici), tedy: `PRODUCTNAME ~obsahuje reg.výraz [0-9]` nebo `PRODUCTNAME ~obsahuje reg.výraz \d`
- Hledáme všechny produkty, které mají v názvu **dvouciferné číslo**, použijeme regulární výraz **[0-9][0-9]** (nebo to samé zajistí použití: `\d\d`)
- Chceme najít všechny produkty, které mají v názvu nějaké **písmeno z následující množiny písmen: b, f, l, m**, použijeme regulární výraz **[bfml]**

### Produkty ?

Jednoduché rozhraní
 Rozšířené rozhraní
 Importovat výběr

PRODUCTNAME ▼
~ obsahuje reg. výraz ▼
[0-9]

## Kulaté závorky ()

Kulaté závorky umožňují vyhledávat v řetězci **různé tvary jednoho slova**, přičemž vyhledají všechny části řetězce, ve kterých se vyskytuje buď část před závorkou nebo celý výraz.

- Chceme najít všechny produkty, které mají v názvu **pneu nebo pneumatika**, použijeme regulární výraz **pneu(matika)?**

### Produkty ?

Jednoduché rozhraní
 Rozšířené rozhraní
 Importovat výběr

PRODUCTNAME ▼
~ obsahuje reg. výraz ▼
sport(ovni)

Prohledávat:  vstup  výstup

## Složené závorky {}

Složené závorky patří mezi kvantifikátory – určují tedy, **kolikrát se dané písmeno nebo řetězec písmen v hledaném textu vyskytuje**. Například:

- Chceme najít všechny produkty, které mají v názvu **písmeno A třikrát za sebou**, použijeme regulární výraz **A{3}**
- Chceme najít všechny produkty, které mají v názvu **písmeno A nejméně jednou**, maximálně dvakrát za sebou, použijeme regulární výraz **A{1,2}**
- Chceme najít všechny produkty, které mají v názvu písmeno **A dva a vícekrát za sebou**, použijeme regulární výraz **A{2,}** - například u módy tak můžeme vytvořit výběr pro největší velikosti (XXL a větší)

The screenshot shows a search interface titled 'Produkty'. It has three tabs: 'Jednoduché rozhraní' (selected), 'Rozšířené rozhraní', and 'Importovat výběr'. Below the tabs is a search bar with a dropdown menu set to 'PRODUCTNAME', a second dropdown menu set to '~ obsahuje reg. výraz', and a text input field containing 'X{1,2}'. At the bottom, there is a 'Prohledávat:' label and two radio buttons: 'vstup' (selected) and 'výstup'.

## Stříška ^

Pomocí této značky vyhledáváme **na začátku řetězce**:

- Chceme najít všechny produkty, jejichž **název začíná číslem** (tj. prvním znakem v názvu produktu je číslo), použijeme regulární výraz **^[0-9] nebo ^\d**
- Chceme najít všechny produkty, jejichž **název začíná mezerou** (tj. prvním znakem v názvu produktu je mezera), použijeme regulární výraz **^\s**
- Chceme najít všechny produkty, jejichž **název začíná slovem Novinka** (tj. v názvu produktu je hned na začátku slovo "Novinka"), použijeme regulární výraz **^Novinka**

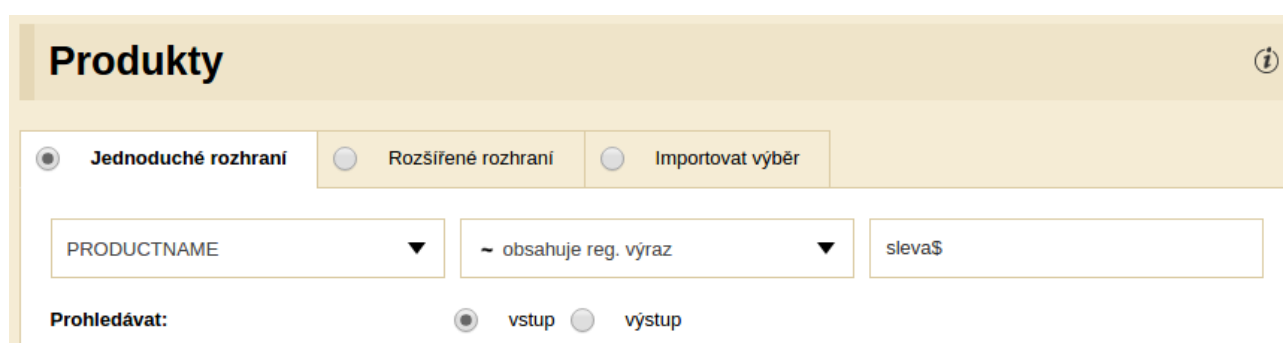
The screenshot shows the same search interface as above. The text input field now contains '^náhradní'. The 'vstup' radio button remains selected.

## Dolar \$

S pomocí "dolaru" vyhledáváme požadované znaky **na konci řetězce**:

- Chceme najít všechny produkty, jejichž **název končí číslem** (tj. posledním znakem v názvu produktu je číslo), použijeme regulární výraz **[0-9]\$ nebo \d\$**
- Chceme najít všechny produkty, jejichž **název končí mezerou** (tj. posledním znakem v názvu produktu je mezera), použijeme regulární výraz **\s\$**
- Chceme najít všechny produkty, jejichž název **končí slovem Novinka** (tj. v názvu produktu je úplně na konci "Novinka"), použijeme regulární výraz **Novinka\$**

Tip: Dolar na klávesnici napíšete např. přes klávesovou zkratku **AltGr a ů**, více způsobů najdete pod tímto odkazem.



**Produkty** ⓘ

Jednoduché rozhraní  Rozšířené rozhraní  Importovat výběr

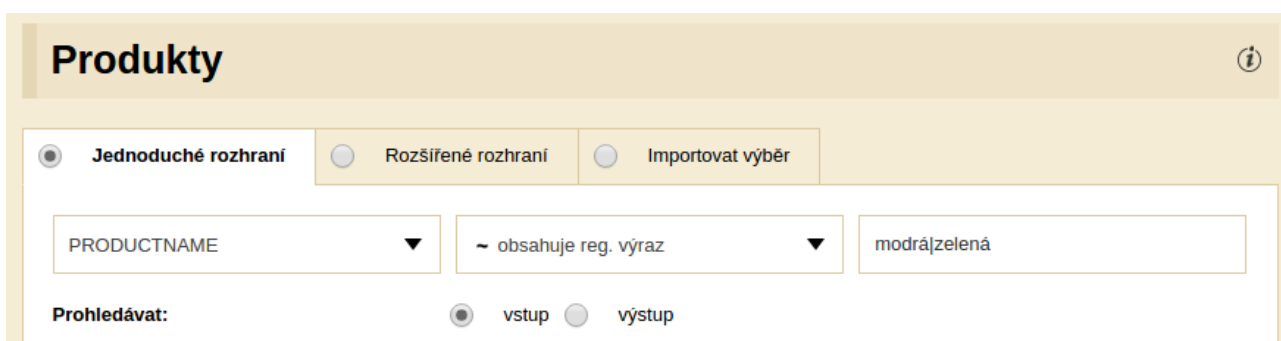
PRODUCTNAME ▼ ~ obsahuje reg. výraz ▼ sleva\$

Prohledávat:  vstup  výstup

## Svislítko |

Tato značka funguje podobně jako operátor OR, česky bychom řekli "nebo" - vyhledá tedy takové položky, ve kterých se vyskytuje **jeden nebo druhý** (třetí, čtvrtý, pátý...) ze zadaných výrazů, tedy:

- pokud chceme najít všechny produkty, jejichž **název obsahuje velikost nebo objem** (tj. kdekoliv v názvu produktu bude jeden nebo druhý hledaný výraz), použijeme regulární výraz **Velikost|Objem**



**Produkty** ⓘ

Jednoduché rozhraní  Rozšířené rozhraní  Importovat výběr

PRODUCTNAME ▼ ~ obsahuje reg. výraz ▼ modrá|zelená

Prohledávat:  vstup  výstup

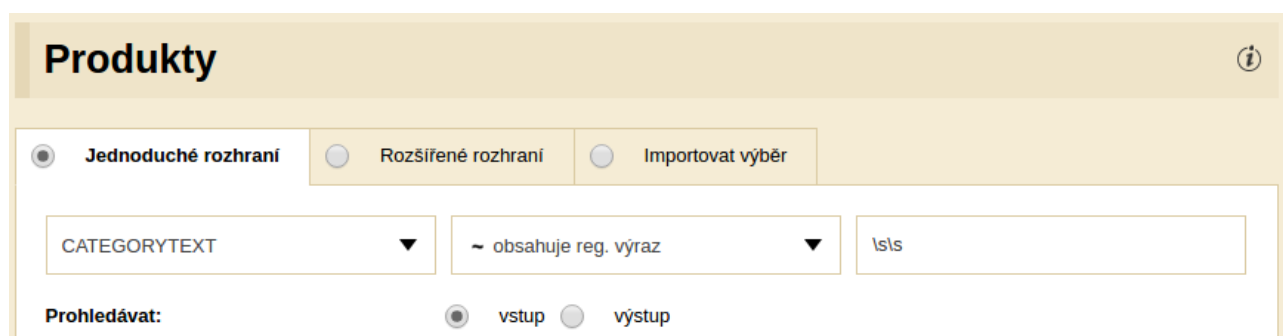


## Znak s se zpětným lomítkem \s

Pomocí tohoto znaku dokážeme vyhledávat, a následně třeba odstranit **mezery (a další bílé znaky** jakou jsou tabulátory nebo nezalomitelné mezery), může se nám tedy hodit při:

- Hledání všech produktů, jejichž **název obsahuje konkrétní víceslovný text**: "Objem<mezera>100<mezera>ml". K tomu použijeme regulární výraz **Objem\s100\sml**
- Odhalování právě **dvou mezer vedle sebe**, kdy použijeme regulární výraz **\s\s**

K vyhledávání **různého počtu mezer** je potom efektivnější použít výraz s kvantifikátorem plus: **\s+**, který odhalí minimálně jednu a maximálně nekonečno mezer za sebou.



The screenshot shows a search interface titled "Produkty". It has three tabs: "Jednoduché rozhraní" (selected), "Rozšířené rozhraní", and "Importovat výběr". Below the tabs are three input fields: "CATEGORYTEXT" with a dropdown arrow, "- obsahuje reg. výraz" with a dropdown arrow, and "lsls". At the bottom, there is a "Prohledávat:" label and two radio buttons: "vstup" (selected) and "výstup".

## Negace

V regulárních výrazech je možné pracovat také s **negací**, a to pomocí **stříšky na začátku množiny znaků**. Například:

- **[^d]** odpovídá jednomu **znaku, který není číslice**,
- **[^s]** odpovídá jednomu **znaku, který není mezera** (nejde o bílý znak).

Regulární výraz **[^s]\*\$** tak bude hledat **jakékoliv znaky od konce řetězce po první mezeru** (bílý znak). Takto je možné najít např. **poslední slovo nebo číslo v řetězci** (viz příklad, ke kterému jsme využili tuto [stránku](#)).

## \D a \S

Při představování jednotlivých značek jsme vynechali ještě dvě důležité a šikovní - **\D** a **\S**. Teď to napravíme:

- **\D** funguje jako **negace malého d**, tedy vyhledá všechny **znaky, které nejsou číslem** a je tedy ekvivalentní výrazu, který jsem představil minule - **[^d]**
- **\S** potom vyhledává takové **znaky, které nepatří mezi bílé**, tedy nejde o mezery, tabulátory, nové řádky apod., taktéž je ekvivalentní výrazu z minula - **[^s]**

## Přehled značek

Pro přehlednost ještě přidávám tabulku všech značek, které v příkladech používáme.

REGULÁRNÍ VÝRAZ	VYHLEDÁ
. tečka	právě jeden neznámý znak
* hvězdička	konkrétní znak 0 až nekonečně krát
? otazník	konkrétní znak 0 nebo 1 krát (znak se nevyskytuje nebo vyskytuje jen jednou)
+ plus	konkrétní znak 1 až nekonečně krát (znak se vyskytuje alespoň jednou)
[] hranaté závorky	jeden ze skupiny hledaných znaků
{ } složené závorky	x opakování znaku nebo skupiny znaků před závorkou
^ stříška	od začátku řetězce
\$ dolar	od konce řetězce
svíslítko	jeden NEBO druhý znak (česky nebo)
\s	mezeru a další bílé znaky (tabulátory, nové řádky)
\d	zastupuje čísla od 0 do 9
\S nebo [^\s]	cokoli kromě bílých znaků, tedy čísla a písmena (negace \s)
\D nebo [^\d]	cokoli kromě čísel, tedy bílé znaky a písmena (negace \d)

# Příklady využití regulárních výrazů v Mergadu

## EAN

EAN může mnohdy způsobovat potíže při zobrazování produktů ve vašem feedu. Podíváme se tedy na dva časté problémy s jeho využitím.

### Hromadné odstranění EAN kódů z popisku produktů

Pro odstranění EANu z DESCRIPTION nebo třeba CATEGORYTEXTU jednotlivých produktů potřebujeme využít pravidlo s regulárním výrazem, které vyhledá číselný řetězec o délce 8-13 znaků, včetně mezery za ním, a nahradí ho za "nic". Využijeme tedy pravidlo typu najít a nahradit, přičemž v elementu CATEGORYTEXT budeme vyhledávat následující regulární výraz: `\d{8,13}\s`, a do pole pro nahrazení nezadáme nic.

Toto pravidlo můžeme aplikovat na všechny produkty, není potřeba dělat speciální výběr - tam kde se EAN v popisku produktu nenajde, nic se nestane a pravidlo se neaplikuje.

### Nové pravidlo

Název:

Výsledná podmínka pro výběr produktů:

Typ pravidla:

Element:

Najít:

Rozlišovat velikost písmen

Regulární výraz

Nahradit:

% Lze zadávat proměnné.

NÁSTROJE

Dostupné proměnné: (43)

### Aplikovat na výběr

Filtrovat:

Všechny produkty (339)

## Kontrola EANu

Chybná délka EANu může být také častým zdrojem potíží třeba při exportu dat na Google. Pomocí regulárních výrazů v Mergadu takovou chybu snadno odladíte.

Řekněme, že chcete vyhledat **všechny produkty, které mají EAN špatně** – obsahuje jiný počet znaků než 8 nebo 12-14 (což jsou nejčastější délky EANu). Budeme tedy vyhledávat takové produkty, které **v elementu EAN nemají právě 8 nebo právě 12-14 znaků**. Tady využijeme stříšku z úvodu, případně v jednoduchém rozhraní Mergada příkaz „**neobsahuje regulární výraz**“:  $^\backslash\{d\{8\}\}\backslash\{d\{12,14\}\}\$$

- **Stříška a dolar** označují začátek a konec řetězce, ve kterém vyhledáváme (oproti předchozímu příkladu tentokrát EAN neobklopuje další text).
- $\backslash d$  vyhledává číslice – buď přesně **8 libovolných číslic** (osmička ve složené závorce) nebo (svislítko) přesně **12 – 14 libovolných číslic**.

### Produkty

Jednoduché rozhraní  Rozšířené rozhraní  Importovat výběr

EAN

Prohledávat:  vstup  výstup

Jen neskryté produkty:

Jen s manuálními změnami:

**HLEDAT**

## Odebrání velikosti z PRODUCTNAME

Některé srovnávače – například Glami – požadují, aby **element PRODUCTNAME neobsahoval informaci o velikosti zboží**, neboť pro ni má speciální element PARAM. V případě, že máte svoje produkty v e-shopu takto pojmenované, potřebujete část názvu smazat u velkého množství položek. Jak na to?

Pomocí **pravidla najít a nahradit** budeme hledat konkrétní regulární výraz: **velikost\s\*\[dSMLX]** + a nahrazovat za "nic".

Chceme totiž najít **slovní spojení velikosti a písmena nebo čísla**, které danou velikost označuje, přičemž mezi slovem „velikost“ a hodnotou může být libovolný počet mezer.

**Nové pravidlo**

**Název:** odstranění velikosti z PRODUCTNAME

**Výsledná podmínka pro výběr produktů:**

**Typ pravidla:** najít a nahradit

**Element:** PRODUCTNAME

**Najít:** velikost\s\*\[dSMLX]

Rozlišovat velikost písmen

Regulární výraz

**Nahradit:**

% Lze zadávat proměnné.

**NÁSTROJE**

**Dostupné proměnné:** (43) @MERGADO\_ID

**Applikovat na výběr**

**Filtrovat:**

Všechny produkty (339)

**VYTVOŘIT** **ZPĚT**

Pojďme ale postupně a ukažme si, jak takový zápis vytvořit:

- **velikost** - hledáme řetězec, ve kterém je slovo „velikost“,
- **hodnota velikosti** – víme, že hodnoty označujeme písmenem (S, M, XL, ...) a občas také kombinací písmena a čísla (např. 4XL). Proto použijeme zápis `\d`, který vyhledává všechna čísla od 0 do 9, a následně **vyjmenujeme všechna písmena, z nichž alespoň jedno se ve výrazu nachází** (proto hranaté závorky),

- toto označení může být kdekoli v řetězci, **nepoužijeme tedy značky pro začátek nebo konec řetězce**,
- mezi slovem "**velikost**" a **samotnou hodnotou může být mezer** (nebo více mezer) a nebo taky nemusí (např. velikostXL, velikost XL), proto použijeme vyhledávání mezer pomocí výrazu `\s`, hvězdička potom říká, že se tam má mezer vyskytovat 0 až nesčítelněkrát,
- znaménko **plus na konci řetězce určuje**, že se celý hledaný výraz v hranatých závorkách vyskytuje v našem řetězci alespoň jednou.

## Složitější příklady využití regulárních výrazů

### Přepsání informace z jednoho elementu do druhého

Častým nedostatkem feedů jsou chybějící nebo špatně naplněné elementy. Podobně často se **požadovaná informace vyskytuje na jiném místě ve vašem kódu** a je tedy logickým řešením ji odtamtud přepsat do správného elementu.

Nejprve je nutné zkontrolovat, že takový **element náš feed obsahuje** a pokud ne, tak ho na kartě **Pravidla doplnit**. Zároveň předpokládejme, že se tento problém týká více než dvou desítek produktů (jinak by bylo snazší řešit jej ručně).

Chceme proto zautomatizovat přepisování tak, aby Mercado poznalo, kde má informaci vzít. Podívejme se na to pomocí příkladu:

Naším záměrem je **naplnit element MANUFACTURER** informacemi o výrobci, která se nachází na začátku PRODUCTNAME.

V prvním kroku tak **vytvoříme proměnnou s názvem MANUFACTURER**, jejíž náplní bude **první slovo z elementu PRODUCTNAME**. (Novou proměnnou přidáte na kartě Pravidla v pravém sloupci skrze příkaz Upravit proměnné).

To zařídíme skrze regulární výraz: `^S+` Tento regulární výraz hledá **od začátku řetězce jakékoliv znaky do první mezery**, najde tedy první slovo v řetězci, a je proto použitelný pro **jednoslovné názvy výrobců**, např. Addidas, Nike, Reebok, atd.

Tuto proměnnou potom vložíme pomocí pravidla Přepsat do elementu MANUFACTURER.

Obdobně můžeme postupovat i u vyhledávání **na jiném místě v konkrétním elementu**:

- první slovo v textu `^S+`
- první dvě slova od začátku textu `^(s*S+){2}`
- poslední slovo na konci textu, včetně případné interpunkce na konci `S+$`
- poslední dvě slova na konci textu, včetně případné interpunkce na konci `(S+s*){2}$`



## Duplicitní slova v elementu

Chceme-li najít **duplicitní slova v jednom elementu**, můžeme také využít regulární výrazy. Poslouží nám pro zpětnou kontrolu dat, a pravidel, která jsme pro jejich správu nastavili.

Například chceme vyhledat **duplicity v PRODUCTNAME**, můžeme takové produkty vyhledat pomocí regulárního výrazu: `(S+)(.*)\1`

Zjednodušeně tímto příkazem Mergadu říkáme: **najdi řetězec znaků (např. slovo), za kterým je libovolný text a následně se taková skupina znaků (tedy např. to slovo) znovu opakuje**. Výraz nebere ohled na interpunkci, najde tedy výrazy, za kterými je čárka i tečka a zároveň i výrazy bez těchto znamének.

Důležité je zmínit, že jde o zjednodušený příklad a takový regulární výraz nebude vyhovovat všem kombinacím, které mohou v textu nastat. Pro naše účely použití v Mergadu však myslím postačí.

V případě, že duplicity skutečně najdete, vyplatí se podívat do pravidel, jakým způsobem se nám tam dostaly (mohla to způsobit například opakující se pravidla).

## Závěrem

K regulárním výrazům ještě považujeme za důležité říct, že jde o široké a obsáhlé téma – v tomto dokumentu jsme popsali jen několik málo způsobů jejich využití. To bylo ostatně také naším cílem – nastínit vám možnosti jejich použití v Mergadu a motivovat vás k jejich používání.

Pokud máte jakékoli další dotazy k regulárním výrazům nebo Mergadu obecně, ozvěte se nám do [fóra](#) nebo přímo na podporu [mergado@mergado.cz](mailto:mergado@mergado.cz).

Nezapomeňte si vyzkoušet test na další straně, který prověří vaše porozumění regulárním výrazům!

## Doporučené zdroje

- [Základy regulárních výrazů](#)
- [Stránka pro zkoušení regulárních výrazů](#)
- [Článek o regulárních výrazech na ABC Linuxu](#)
- [Článek o regulárních výrazech na root.cz](#)
- [Anglický web o regulárních výrazech](#)



# Test na závěr

Vytvořte takový regulární výraz, kterým:

1. najdete všechna slova delší, než 5 znaků
2. najdete trojčíferné číslo v řetězci
3. najdete poslední slovo na konci textového řetězce
4. najdete text, který začíná slovem "https://"
5. najdete text, který neobsahuje žádné bílé znaky (např. mezery)
6. zkontrolujete, že poslední znak v řetězci je tečka .
7. zkontrolujete formát telefonního čísla, tj. 9 čísel, oddělené mezerami po třech číslech (123 456 789)
8. zkontrolujete, že celý řetězec je maximálně 255 znaků dlouhý
9. odkážete na první skupinu z regulárního řetězce
10. zkontrolujete, že text neobsahuje čísla, bílé znaky, ani znak "@"



## Výsledky testu

1. např. `\S{6,}` nebo `[\^s]{6,}` (jde více způsoby), kdybychom hledali slova delší nebo rovna 5 znakům, pak by ve složených závorkách bylo číslo 5
2. `\d{3}` nebo `[0-9][0-9][0-9]` nebo `\d\d\d` zachytí 3 a více ciferná čísla, pokud potřebujeme přesný počet cifer, musíme to nějak ohraničit, např. `^\d{3}$` jako celý řetězec nebo `\s\d{3}\s` pokud předpokládáme, že je číslo obklopeno mezerami nebo použijeme značku `\b` pro vymezení hranice slova a v našem seriálu se na ni nedostalo místa: `\b\d{3}\b`
3. `\S+$` nebo `[\^s]+$`
4. `^https:\V` protože lomítka jsou v rámci regulárních výrazů speciální znaky, je třeba je ošetřit zpětným lomítkem
5. `^\S+$` nebo `^\[\^s]+$` prostě od začátku do konce řetězce nesmí být mezera, proto využijeme značek pro začátek `^` a konec `$` řetězce
6. `\.$` protože je tečka speciální znak, který má v regulárních výrazech konkrétní funkci, je třeba ji doplnit o zpětné lomítko, pokud ji chceme vyhledat jako opravdový znak tečky
7. `\d{3}\s\d{3}\s\d{3}` nebo kratší zápis `(\d{3}\s*){3}` by teoreticky stačil, není zde však ošetřen výskyt víceciferných čísel, takže stejně jako v odpovědi u bodu 2) použijeme značky pro hranice a výsledkem bude `(\b\d{3}\b\s*){3}`
8. `^\[S\s]{0,255}$` tedy jakékoliv znaky (písmena a čísla nebo mezery) se budou opakovat nula až 255 krát
9. `\1`
10. `^\[^\d\s@]+$` tedy od začátku do konce řetězce může být cokoliv, co není číslo nebo bílý znak nebo znak `@`